

IMPLEMENTASI ALGORITMA *EDIT DISTANCE* PADA PENGEMBANGAN APLIKASI E-LEARNING BSI MENGUNAKAN METODOLOGI *WATERFALL*

Iqbal Dzulfikar Iskandar¹, Taufiqurrochman²

¹Manajemen Informatika, AMIK BSI, Tasikmalaya, Jl. Tanuwijaya No.4, Empangsari, Tawang, Tasikmalaya,

^{1,2} Magister Ilmu Komputer, STIMIK Nusa Mandiri, Jakarta

iqbal.iql@bsi.ac.id

Abstrak

Aplikasi E-learning Bina Sarana Informatika dipergunakan sebagai sarana untuk mempermudah dosen dan mahasiswa untuk melakukan proses belajar mengajar dengan mode tanpa bertatap muka. Aplikasi E-learning, memiliki konten yang berisi seputar pengolahan data akademik, baik itu proses pembuatan latihan soal oleh dosen, tugas, quis, *upload* modul, serta penilaian hasil belajar mahasiswa. Berkaitan dengan hal tersebut kekurangan yang terdapat pada aplikasi *E-learning* BSI adalah tidak memiliki fitur untuk pencarian modul diktat mahasiswa. Maka dari itu perlu di kembangkan aplikasi *E-learning* BSI dengan menambahkan fitur proses pencarian modul diktat kuliah mahasiswa, yang bertujuan untuk mempermudah proses temu kembali informasi dalam pencarian modul diktat pembelajaran mahasiswa. Proses pengembangan perangkat lunak pada *e-learning* BSI menggunakan pendekatan model proses *waterfall* dengan mengadaptasi algoritma *edit distance* ke dalam fitur pencarian modul diktat pada *website e-learning* BSI. Hasil pengujian pencarian modul dengan algoritma *edit distance* menggunakan kata *typo* “dta minig”, “algoritma genetic”, “J4va web”, “racle”, dapat dikoreksi dan terdeteksi oleh algoritma *edit distance* dengan akurasi 90% dengan tingkat kecepatan pencarian 48% lebih cepat dari pencarian yang tidak mengadopsi algoritma *edit distance*, ini mengartikan algoritma *edit distance* dapat mengkoreksi kata pada saat pengguna salah dalam pengetikan kata.

Kata kunci: E-learning, BSI, *Edit Distance*, *Waterfall*, *Website*

Abstract

E-learning application Bina Sarana Informatika is used as a means to spread and learn to learn in a fashion without face to face. E-learning application, contains content that contains about the processing of academic data, both the process of making assignments by lecturers, assignments, quizzes, upload modules, and also student learning outcomes. The disadvantage found in the BSI E-learning application is that it does not have a feature to search for student dictate modules. Therefore, it is necessary to develop the BSI E-learning application by adding a search process feature of the student's lecture module, which aims to facilitate the information retrieval procession in the search for student learning dictators. The development process carried out refers to the software engineering methodology with the waterfall

Keywords : E-learning, BSI, *Edit Distance*, *Waterfall*

PENDAHULUAN

Aplikasi E-learning berbasis *website* pada perguruan tinggi merupakan inovasi pembelajaran yang dilakukan tanpa harus bertatap muka atau dapat di sebut dengan pembelajaran dengan *Mode Online*, proses E-learning dilakukan ketika dosen dan mahasiswa memiliki cukup waktu untuk berkomunikasi di

luar jam kuliah efektif. Baik berdiskusi seputar matakuliah yang di ampu oleh dosen yang bersangkutan, penelitian, share modul kuliah, pemberian tugas dan penilaian tugas kepada mahasiswa melalui *mode E-learning*. Dengan aplikasi *E-learning* mahasiswa dapat mengerjakan tugas kuliah secara *online*, dan melakukan komunikasi melalui *chating* dengan

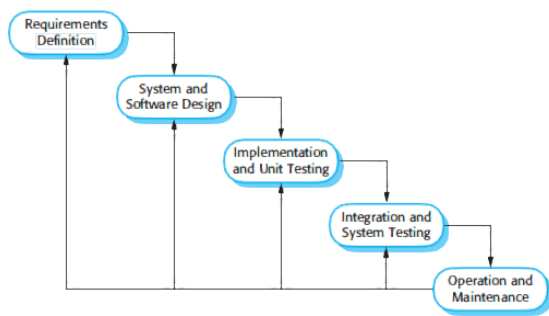
dosen, sehingga mahasiswa dapat tetap belajar dimanapun dan kapanpun. Salah satu contohnya *E-learning* berbasis *website* yang diterapkan oleh Stimik BSI kepada dosen dan mahasiswa untuk proses Mengajar belajar, dalam pengaplikasiannya, dosen dapat melakukan serangkaian proses akademik secara online melalui *website E-learning* BSI, baik itu proses penambahan modul diktat, absensi siswa, sampai pemberian tugas dan penilaian mahasiswa, begitu juga dengan mahasiswa, dapat mengakses materi-materi yang diberikan oleh dosen pengampu. Aplikasi *E-learning* yang digunakan BSI memiliki banyak fitur yang memudahkan penggunaannya, tetapi tentu saja masih terdapat sedikit kekurangan yang terdapat pada proses temu kembali informasi, yaitu tidak adanya fitur yang mendukung untuk proses pencarian modul diktat kuliah, sehingga mahasiswa hanya dapat mengakses modul yang ditampilkan saja di aplikasi dengan matakuliah yang diambil. Maka dari itu pada penelitian ini akan dilakukan perancangan yang mengacu kepada model Metodologi *Software engineering* pengembangan *system*, *System Development Life Cycle*, dengan model penelitian waterfall, metode ini di gunakan untuk mengembangkan aplikasi *E-learning* yang berbasis temu kembali informasi di lingkungan Bina Sarana Informatika(BSI), sedangkan algoritma yang dipergunakan untuk proses implentasi aplikasi pada E-learning BSI menggunakan algoritma *Edit distance*.

Algoritma *edit distance* merupakan algoritma yang dipergunakan untuk proses pengkoreksian string, seperti menghapus, mengubah dan menyisipkan string menggunakan perhitungan nilai terkecil dengan melalui matriks. Iyan. Mulayana (2015) pada penelitiannya yang berjudul Identifikasi Plagiasi Karya Ilmiah berbasis Temu Kembali Informasi Menggunakan Algoritma *Edit Distance* Melalui Peringkasan Teks Otomatis menyimpulkan “Peringkasan teks otomatis mempengaruhi tingkat kecepatan proses pencarian sebesar 48%”. Selain itu algoritma *Edit distance* dapat dipergunakan sebagai pengkoreksi kata pada saat pengguna melakukan kesalahan pengetikan ketika pencarian berbasis temu kembali informasi. Seperti pada penelitian Ni Made Muni Adriyani, I Wayan Santiyasa, Agus Muliantara (2016) yang berjudul “Implementasi Algoritma *Levenshtein Distance* Dan Metode

Empiris Untuk Menampilkan Saran Perbaikan Kesalahan Pengetikan Dokumen Berbahasa Indonesia” menyimpulkan “Dengan menerapkan Algoritma *Levenshtein Distance* dapat membantu mengatasi permasalahan pada kesalahan pengetikan dengan mekanisme penambahan, penyisipan dan penghapusan karakter”. Berdasarkan teori yang telah dikemukakan. Maka dapat disimpulkan untuk pengembangan aplikasi *E-learning* Binasarana Informatika dapat dilakukan dengan menggunakan algoritma *edit distance* untuk pembuatan fitur proses pencarian modul kuliah pada halaman *website*.

METODE

Metode *Software Engineering* merupakan pendekatan terstruktur dan sistematis terhadap pengembangan perangkat lunak yang bertujuan memfasilitasi produksi perangkat lunak kualitas tinggi dengan cara yang efektif dalam hal biaya (Ian Sommerville, 2011), dalam metode *Software Engineering* terdapat model pengembangan *system* yang disebut dengan model *waterfall*, *waterfall* merupakan model pengembangan perangkat lunak yang sistematis atau dapat dikatakan sebagai siklus hidup perangkat lunak, proses tahapan yang harus dilakukan dengan model *waterfall*, dari mulai perancangan sampai software yang telah dibangun di terapkan pada *system* yang berjalan. Tahapan-tahapan yang dapat dilakukan meliputi lima tahapan, tahap pertama yaitu Analisis dan definisi persyaratan, setelah tahap satu selesai. Maka tahap dua dapat di proses yaitu Perancangan *system* dan perangkat lunak, setelah proses dua selesai, proses berikutnya masuk ke siklus tiga yaitu Implementasi dan pengujian unit, proses selanjutnya masuk kedalam siklus empat yaitu Integrasi dan pengujian *system*, fase terakhir adalah Operasi dan pemeliharaan. Masing-masing siklus memiliki proses yang berbeda tetapi saling berkaitan antara siklus yang satu dengan yang lainnya.



Gambar 1. Model *Waterfall* (Ian Sommerville, 2011).

Gambar 1 (Ian Sommerville, 2011) menunjukkan proses yang harus dilakukan ketika mengembangkan *software* menggunakan model *waterfall*. Penjelasan dari setiap tahapan pada gambar 1 model *waterfall* adalah sebagai berikut:

1. *Requirements Definition*

Pelayanan, batasan, dan tujuan system ditentukan melalui konsultasi dengan user system. Persyaratan inikemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi system.

2. *System And Software Design*

Proses perancangan system membagi persyaratan dalam system perangkat keras atau perangkat lunak. Proses ini menentukan arsitektur system secara keseluruhan, mulai dari abstraksi sampai hubungan yang berelasi atau berkait.

3. *Implementation and unit testing*

Pada tahap ini, adalah proses dari menerjemahkan perancangan kedalam serangkaian program atau unit program. Pengujian dilakukan untuk meverifikasi setiap unit dapat bekerja dengan baik dan sesuai dengan spesifikasinya.

4. *Integration And System Testing*

Unit program individual diintegrasikan dan diuji sebagai system yang lengkap untuk menjamin bahwa persyaratan sistem telah terpenuhi. Setelah pengujian system, perangkat lunak dikirim kepada pelanggan.

5. *Operation And Maintenance*

Tahap ini adalah fase siklus hidup yang paling lama. Karena masuk kedalam proses penggunaan, sehingga perangkat lunak harus tetap dilakukan perawatan secara berkala, mencakup koreksi dari berbagai *error* yang tidak ditemukan pada tahap-tahap terdahulu.

Lalu penambahan fitur pada perangkat lunak jika dibutuhkan.

Metode *Software Engineering* pada umumnya memiliki karaktersitik yang kompleks, yaitu dipergunakan untuk membangun *software* yang sangat besar, tentunya harus melibatkan setiap sub fungsi sehingga dapat dikerjakan ditempat berbeda. Berkaitan dengan karakteristik tersebut. Maka metode *Software Engineering* dengan model *Waterfall* dapat di implemantasikan pada proses pengembangan aplikasi E-Learning BSI berbasis temu kembali informasi.

Algoritma *Edit Distance*

Algoritma *Edit distance* adalah algoritma yang sering dipergunakan untuk mengubah, menghapus serta menyisipkan string menggunakan perhitungan matriks, proses perhitungannya dimulai dari nilai awal yang terkecil lalu di proses sampai perhitungan selesai. Dari hasil perhitungan tersebut akan didapat nilai akhir yang akan menunjukkan jumlah operasi berapa kali kalimat tersebut dilakukan perubahan. Pada awalnya Algoritma *Edit Distance* ditemukan oleh Vladimir Levinshatein pada tahun 1965, sehingga di beri nama Algoritma *Levinshatein distance*. Operasi yang dilakukan oleh algoritma *Edit Distance* pertama operasi pengubahan karakter string, operasi penambahan karakter, operasi penghapusan karakter.

1. Operasi pengubahan karakter

prosesnya adalah mengubah karakter string yang salah akibat dari kesalahan pengetikan, contoh kalimat “pulajg”, dalam Bahasa Indonesia kalimat “pulajg tidak memiliki arti, sehingga dengan metode Edit Distance kalimat “pulajg” dapat diubah menjadi kalimat “pulang”. Prosesnya string huruf “j” diubah menjadi string huruf “n”. Sehingga kalimat akan memiliki arti.

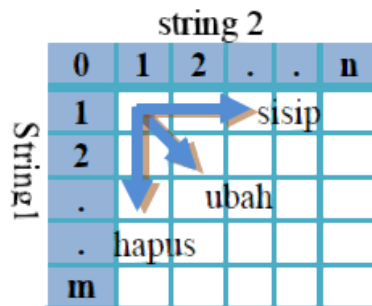
2. Operasi Penambahan Karakter

Proses ini dilakukan jika user mengetikan kalimat yang tidak sempurna, contoh seperti kalimat “Bers”. Maka dengan proses penambahan karakter, karakter akan ditambahkan huruf “a” sehingga menjadi “Beras”.

3. Operasi penghapusan karakter

Operasi penghapusan karakter dilakukan ketika string yang diketikan oleh pengguna melebihi

kalimat yang dimaksudkan sehingga kalimat tersebut memiliki arti yang lebih relevan, contoh “sayangs”. Maka string “s” akan dihapus dari kalimat, sehingga menjadi “sayang”.



Gambar 2. Matriks nilai *edit distance* (Samsu sampena, 2009)

Keterangan:

S1 = String 1

S2 = String 2

Rumus yang digunakan untuk perhitungan tabel adalah sebagai berikut:

$$(n+1) \times (m+1) \quad (1)$$

n = panjang string ke 1

m = panjang string ke 2

S1 merupakan variabel untuk string yang akan diperbaiki, yang nantinya dibandingkan dengan string S2, S2 adalah string yang digunakan sebagai referensi S1 ketika proses perbandingan dan perubahan dilakukan.

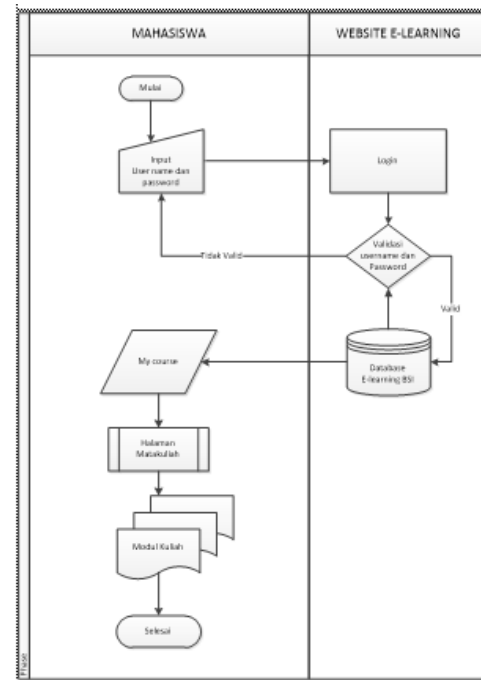
Contoh *string* kalimat yang salah “IQBXL”, akan diperbaiki menjadi “IQBAL” menggunakan perhitungan *matriks edit distance*. Dalam kasus yang lebih *kompleks*, string kalimat yang salah akan dibandingkan dengan jutaan bahkan triliunan kata yang terdapat didalam database, sehingga sistem akan memberikan berbagai macam kata yang sama atau hampir sama dengan kata tersebut.

Tabel 1. Tabel *Edit Distance*

	I	Q	B	A	L
Q	1	0	1	2	3
B	2	1	0	1	2
A	3	2	3	0	1
X	4	3	4	1	0
L	5	4	4	2	1

Tabel 1 menunjukkan hasil akhir perhitungan matriks Edit Distance bernilai satu, sehingga

diketahui hanya dibutuhkan satu kali proses perubahan untuk memperbaiki kalimat “IQBXL” menjadi “IQBAL”, proses yang digunakan yaitu proses hapus karakter “X”.

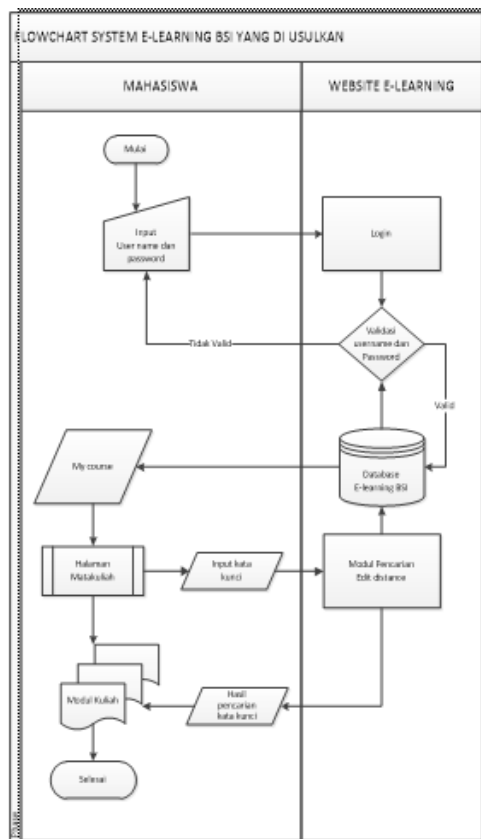


Gambar 3. Flowchart Sistem Website E-learning BSI yang berjalan

Flowchart system pada gambar 3 adalah proses yang dilakukan oleh mahasiswa ketika login sampai masuk kedalam pembelajaran. Prosesnya pertama mahasiswa melakukan login terlebih dahulu kedalam *system*, jika username dan password benar. Maka *system* akan memvalidasi dan masuk kedalam halaman utama ruang mahasiswa. Proses kedua, mahasiswa memilih matakuliah yang sudah ada pada halaman ruang crouse, selanjutnya dapat mengunduh modul yang diperlukan sesuai dengan matakuliah yang akan dipelajari.

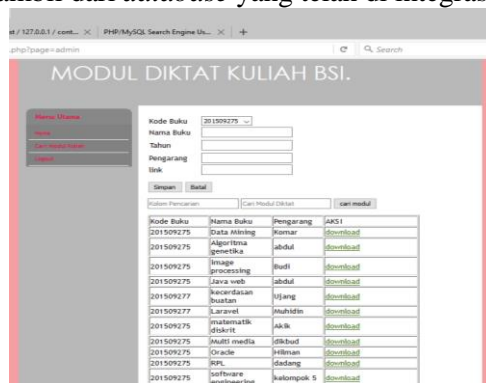
HASIL DAN PEMBAHASAN

Proses implementasi dilakukan setelah proses analisa dan tinjauan penelitian website E-learning dilakukan. Untuk mengintegrasikan algoritma *edit distance* pada proses pencarian modul diktat matakuliah mahasiswa *website E-learning* Bina Sarana Informatika perlu dilakukan perancangan terhadap alur sistem, tujuan perancangan ini adalah untuk meninjau serta membuat konsep sistem yang akan diterapkan sebelum masuk kedalam tahap implementasi yang sebenarnya.



Gambar 4. Flowchart Sistem yang di Usulkan

Gambar 4 menunjukkan flowchart sistem yang di usulkan sebagai proses pengembangan dari aplikasi *website E-Learning* Bina Sarana Informatika. Pengembangan yang dilakukan yaitu menambahkan proses pencarian modul diktat. Dengan begitu mahasiswa dapat mendapatkan berbagai macam modul yang di peroleh dari hasil pencarian. Referensi modul diambil dari *database* yang telah di integrasikan.

Gambar 5. Tampilan awal *prototype* halaman modul diktat BSI dengan kolom pencarian *edit distance*

Pengujian dilakukan dengan memasukkan kata kunci yang salah kedalam kolom pencarian

No.	Kata Kunci	Terdeksi	Tidak Terdeteksi
1	Dta Mning	v	
2	Algoritma genetika	v	
3	racle		v
4	J4va web	v	

Hasil pengujian yang dilakukan, *algoritma edit distance* dapat mendeteksi kata yang terdapat pada *database* meskipun kalimat tersebut salah atau tertinggal dalam pengetikan. Pengujian pencarian modul dengan algoritma *edit distance* menggunakan kata *typo* “dta minig”, “algoritma genetic”, “J4va web”, “racle”, dapat dikoreksi dan terdeteksi oleh algoritma *edit distance* dengan akurasi 90% dengan tingkat kecepatan pencarian 48% lebih cepat dari pencarian yang tidak mengadopsi algoritma *edit distance*, ini mengartikan *algoritma edit distance* dapat mengkoreksi kata pada saat pengguna salah dalam pengetikan kata. Tetapi masih terdapat kekurangan, yaitu sensitifitas hasil pencarian yang kurang, hal ini menjadikan mesin pencarian tidak dapat mendeteksi kalimat yang mirip atau berdekatan. Untuk dapat mengkoreksi kata yang salah algoritma *edit distance* harus membandingkan terlebih dahulu dengan kata yang benar yang terdapat pada *database*. Sehingga semakin banyak kalimat yang terdapat pada *database*. Maka algoritma akan semakin sensitif dalam mendeteksi kata yang dicari.

SIMPULAN DAN SARAN

Implementasi algoritma *edit distance* pada aplikasi *website E-learning* Bina Sarana Informatika dapat dilakukan. Dengan mengacu kedalam metodologi pengembangan *system* yaitu *software engineering* model *waterfall*.

Pengembangan dilakukan terhadap penambahan fitur pencarian modul diktat mahasiswa, berbasis temu kembali informasi. Dengan adanya fitur pencarian modul. Mahasiswa dapat mendapatkan berbagai macam referensi modul diktat yang di butuhkan tanpa harus mengambil terlebih dahulu matakuliah atau masuk kedalam halaman pembelajaran (crouse). Tetapi kekurangan dari algoritma *Edit Distance* adalah kurang nya sensitifitas dalam mendeteksi kata yang terdapat pada *database*. Untuk memperbaiki hal tersebut, *user* harus memasukkan banyak kalimat kedalam *database*.

Sehingga algoritma akan semakin sensitif dalam mendeteksi kata yang di cari.

Implementasi algoritma edit distance pada system informasi aplikasi E-learning BSI yang telah berjalan telah dilakukan. Tetapi terdapat beberapa pengembangan system yang harus lebih di sempurnakan lagi pada penelitian berikutnya. Beberapa saran yang berkaitan dengan kekurangan pada penelitian ini diantaranya.

Pengembangan system hanya dilakukan didalam lingkup halaman mahasiswa saja. Sehingga untuk penelitian selanjutnya, pengembangan proses pencarian diktat dapat dilakukan pada halaman dosen.

Proses pencarian tidak hanya mencakup modul diktat mahasiswa saja, tetapi dapat juga dikembangkan dengan menambahkan fitur pencarian yang lebih luas, seperti pencarian informasi tentang dosen pengampu matakuliah tertentu dan mahasiswa.

DAFTAR PUSTAKA

- Adriyani, M. M. N., Santiyasa, W. I., Muliantara A. (2011). *Implementasi Algoritma Levensthein Distance dan Metode Empiris Untuk Menampilkan Saran Saran Perbaikan Kesalahan Pengetikan Dokumen*. Universitas Udayana.
- Baskoro SY., Ridok , Furqon TM., 2015. *Pencarian Pasal Pada Kitab Undang-Undang Hukum Pidana (Kuhp) Berdasarkan Kasus Menggunakan Metode Cosine Similarity Dan Latent Semantic Indexing (Lsi)*. Journal of Environmental Engineering & Sustainable Technology JEEST. P-ISSN:2356-3109.
- Fika HR. 2012. *Sistem Temu Kembali Informasi Menggunakan Model Ruang Vektor dan Inverted Index*. Madura. Jurnal Simantec. Fakultas Teknik, Universitas Trunojoyo Madura. ISSN 2088-2130.
- Hamzah A. 2009. *Temu Kembali Informasi Berbasis Kluster Untuk Sistem Temu Kembali Informasi Teks Bahasa Indonesia*. Fakultas Teknologi Industri Institut Sains & Teknologi AKPRIND. Yogyakarta. Jurnal Teknologi, Volume 2 Nomor 1.
- Karter D P., Lumenta A., Jacobus A. 2016. *Penerapan Sistem Temu Kembali Informasi Pada Kumpulan Dokumen Skripsi*. E-journal Teknik Informatika, Volume 8. ISSN : 2301-8364
- Mulyana, I., Chairunnas, A., Maesya A. (2015). *Identifikasi Plagiasi Karya Ilmiah berbasis Temu Kembali Informasi Menggunakan Algoritam Edit Distance Melalui Peringkasan Teks Otomatis* Jurnal Nasional. Universitas Pakuan. ISSN 2407 – 7534.
- Magdalena M.G. Lum. 2015. *Pemanfaatan Sarana Temu Kembali Informasi Khususnya Katalog Upt Perpustakaan Unima Oleh Mahasiswa*. Journal “Acta Diurna” Volume IV. No.1.
- Muhammad, Endah N.S., Noranita B. *Sistem Temu Kembali Informasi Dalam Dokumen Menggunakan Metode Latent Semantic Indexing*. Program Studi Teknik Informatika. Universitas Diponegoro.
- Sempena, S. (2009). *Algoritma Program Dinamis Edit distance untuk Pengecekan Ejaan*. Institut Teknologi Bandung.
- Sommerville. (2011). *Software Engineering* 9th editon. Pearson Education, Inc. Boston
- Piliang M. *Sistem Temu Kembali Informasi Dengan Mendayagunakan Media Katalog Perpustakaan*. Jurnal Iqra' Volume 07 No.02.
- Witanti W., Rahmanto H., Renaldi F. 2016. *Pembangunan Sistem Temu Balik Informasi (Information Retrieval) Dalam Pemilihan Pemain Sepak Bola Berkualitas Di Indonesia Berbasis Analisis Sentimen*. Seminar Nasional Teknologi Informasi dan Komunikasi (SENTIKA 2016). Yogyakarta ISSN: 2089-9815.